

CA ERwin[®] Model Manager

Implementation Guide

r7.3



This documentation and any related computer software help programs (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of the Documentation for their own internal use, and may make one copy of the related software as reasonably required for back-up and disaster recovery purposes, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the provisions of the license for the Product are permitted to have access to such copies.

The right to print copies of the Documentation and to make a copy of the related software is limited to the period during which the applicable license for the Product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

EXCEPT AS OTHERWISE STATED IN THE APPLICABLE LICENSE AGREEMENT, TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

The use of any product referenced in the Documentation is governed by the end user's applicable license agreement.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Copyright © 2008 CA. All rights reserved.

CA Product References

This document references the following CA products:

- CA ERwin® Model Manager (CA ERwin MM)
- CA ERwin® Data Modeler (CA ERwin DM)
- CA ERwin® Process Modeler (CA ERwin PM)
- CA ERwin® Model Navigator (CA ERwin MN)

Contact CA

Contact Technical Support

For your convenience, CA provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA products. At <http://ca.com/support>, you can access the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Provide Feedback

If you have comments or questions about CA product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA product documentation, complete our short [customer survey](#), which is also available on the CA support website, found at <http://ca.com/support>.

Contents

Chapter 1: Modeling in the Multi-User Environment	9
Workgroup Model Management Systems	9
Model Life Cycle Frameworks	11
Model-Driven Development Framework	11
System-Driven Model Framework	12
Informational Model Framework	12
Business Process Model Framework	13
 Chapter 2: Installation and Configuration	 15
Prepare for the Installation	15
Specify Use of Foreign Characters With Microsoft SQL Server 2000	16
Specify Use of Foreign Characters With Microsoft SQL Server 2005	16
How to Prepare Your DBMS Environment	17
Tasks to Create a Microsoft SQL Server DBMS	18
Tasks to Create a Sybase DBMS	19
Transact SQL Commands	21
Tasks to Create an Oracle DBMS	22
CA ERwin MM Mart Creation Requirements	28
Server Connection Software	28
Connection Prerequisites	29
Install the Administrative Files	30
License CA ERwin Model Manager	31
Refresh the Mart After Upgrading Your License	32
Create the CA ERwin MM Mart	33
 Chapter 3: Upgrade	 39
Software Upgrades	39
Conversion Process	39
Property Conversion	40
Model Property Conversion	40
Conversion Methods	40
How to Upgrade the Mart	42
Microsoft SQL Server or Sybase DBMS Upgrade Considerations	43
Oracle DBMS Upgrade Considerations	43
CA ERwin MM 4.1.4 to CA ERwin MM r7.3 Conversion	43
Install the Software Upgrade	44

Convert or Update a CA ERwin MM Mart	46
--	----

Appendix A: Microsoft SQL and Sybase Server Tuning Recommendations 51

Physical Tuning Parameters	51
Hardware Configurations	51
Database Placement	52
Data Redundancy	52
Server-Level Tuning Parameters	52
Named Cache Configuration	53
Database Tuning Parameters	53
Transaction Log	53
Threshold Procedures	54
"trunc. log on chkpt" Option	54
Database and Log Sizing	55
How to Maintain an Efficient Database	55
Nightly Maintenance Script	56
DBCC Commands	57

Appendix B: Oracle Tuning Recommendations 59

Recommended Disk Configurations	59
Database Configuration	60
Maximum Number of Processes	60
Shared Pool	60
Buffer Cache	61
Redo Log Files	61
Increase Space in Redo Logs	62
Redo Log Buffer	64
Checkpoints	65
Turn on the Optimizer Mode	65
Storage Parameters	66
Rollback Segments	66
Set Optimal Configuration for the Rollback Logs	67
Tablespaces	69

Appendix C: CA ERwin MM Performance 71

Optimize Your CA ERwin MM Performance	71
Reindex the Database	72
Run the Oracle DBMS Reindex Script	73
Example: Oracle DBMS Reindex Script	74
Example: Microsoft SQL Server DBMS Reindex Script	75

Example: Sybase DBMS Reindex Script	76
Troubleshooting the Reindexing Script	77
Run the Database Statistics	77
Example: Statistics Commands (Oracle)	77
Example: Statistics Script (MS SQL Server)	78
Index	79

Chapter 1: Modeling in the Multi-User Environment

CA ERwin MM coordinates the development and management of models created with CA ERwin Data Modeler and CA ERwin Process Modeler.

This section contains the following topics:

[Workgroup Model Management Systems](#) (see page 9)

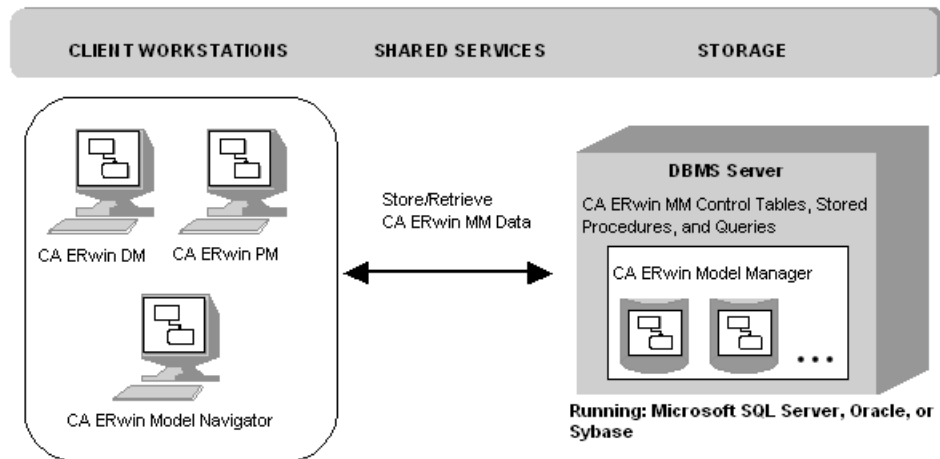
[Model Life Cycle Frameworks](#) (see page 11)

Workgroup Model Management Systems

CA ERwin MM provides a multiuser modeling environment that makes coordinated, large-scale modeling possible. It enables collaboration among project managers, data modeling team members, process modeling team members, and standards administrators by providing workgroup modeling services, including conflict resolution, versioning, security, and standards management. You can coordinate the efforts of model teams to document existing systems, create new eBusiness systems, and drive data standardization. Model sharing encourages teamwork, so modelers can work together more efficiently to optimize model-based development. Your modelers get more work done in less time with better results.

Workgroup modeling operates in a client/server environment so processing is shared between the user client workstation and the server where the models are stored. Users access the stored models through one of the clients-CA ERwin Data Modeler or CA ERwin Process Modeler. Models are held in a platform and network independent database or mart that resides on a central server, using Microsoft SQL Server, Oracle, or Sybase as the host database.

The following diagram shows a typical CA ERwin MM environment:



Workgroup modeling features are provided to help control updates to models when you are opening, closing, and saving models. The administrator installs the program and initializes the database on the DBMS server. The administrator also has the responsibility of setting up the library structure in which models are organized, and assigning security profiles to users. Administration overhead is relatively low and proportional to the number and activity level of the users. After a model is saved to the database, control of who can work on the model and how changes are saved to the model is handled by security, making it possible for workgroups to work on large models without confusion. CA ERwin MM supports many security and administrative features related to its multiuser capabilities and the client/server environment in which it operates. For this reason, routine management is usually performed by a dedicated administrator. Administrators can find detailed information regarding their tasks and responsibilities in the *Administration Guide*.

You can also use CA ERwin Model Navigator as a client. Read-only access is permitted to data models and process models, so your workgroup can use CA ERwin Model Navigator to share information with others without the risk that unauthorized changes might be saved to the mart. Additionally, you can connect to the mart through the Naming Standards Editor or the Datatype Standards Editor.

Model Life Cycle Frameworks

Use one of the following model life cycle frameworks in your organization:

Model-Driven Development

Changes to the schema are made to the model first and then forward engineered.

System-Driven Models

Changes are made directly to the schema and the schema is reverse-engineered into the model to reflect the changes.

Informational Models

Contains logical-only models, enterprise-wide models, or standards and sample models.

Business Process Models

Contains models generated by CA ERwin Process Modeler.

Each type of framework has different considerations that you must think about when configuring your database and developing its supporting policies. You are not required to choose a particular framework, however, it helps to know your development process before building a library structure.

Model-Driven Development Framework

In the model-driven framework, the model is always the source of all changes. You create a new database schema by forward engineering the model. The life cycle of a model in the model-driven Development framework can follow a path like this:

- Create the library structure (for example, Development, Test, and Production) and populate them with CA ERwin Data Modeler templates.
- Create the logical model in a development library.
- Promote the model to the test library when it is ready.
- Generate the schema from the test library.
- Modify the test model as required and synchronize it to the schema.
- Promote the model to the production library when it is ready.
- Publish the refreshed production model.
- Update the enterprise-wide model, if necessary.
- Incorporate changes into the development model for further changes, and repeat the process.

System-Driven Model Framework

In the system-driven framework, there is an established information system from which you can reverse engineer database tables. The life cycle of a model in the system-driven Development framework can follow a path like this:

- Create the library structure (for example, Reverse Eng, Test, Production). You should not require CA ERwin Data Modeler templates because you do not create models from scratch.
- Reverse engineer the model from the information system into the designated library.
- Enhance the model with logical information and input from analysts.
- Update the model to reflect changes in the physical schema.
- Create a version of the model.
- Synchronize the schema and the model using Complete Compare.
- Publish the model.
- Repeat the last three steps as the system is modified.

Informational Model Framework

In the Informational Model framework, CA ERwin MM contains CA ERwin Data Modeler logical-only models, enterprise-wide models, or standards and sample models. There is no forward engineering with the intent of using the schema. The life cycle of a model in the Informational Model framework can follow a path like this:

- Create the library structure and populate them with CA ERwin Data Modeler templates.
- Develop the initial model. Use reverse engineering and model new components as required.
- Publish the initial model to the appropriate parties for modification and refinement.
- Get approval, and then version the model.
- Publish the approved model.
- Update and publish models as the enterprise model evolves.

Business Process Model Framework

In the Business Process Model framework, you use CA ERwin Process Modeler to design business process models that define business processes in your organization. You can optionally export entities and attributes to CA ERwin Data Modeler to develop a model-driven information system that adheres to the business process rules. The life cycle of a model in the Business Process Model framework can follow a path like this:

- Create a working version of an AS-IS model of selected subjects.
- Critique and analyze the model.
- Move the model through draft, recommended, and publication versions.
- Publish the model at the end of the cycle.
- Create working TO-BE models of the model from Step 4 that incorporates management's future vision.
- Update the model as changes and enhancements are approved and as support systems are integrated into the models.
- Version the modified models based upon the approval of the changes and enhancements.
- Publish the revised models when each version becomes stable.

Chapter 2: Installation and Configuration

This section contains the following topics:

[Prepare for the Installation](#) (see page 15)

[How to Prepare Your DBMS Environment](#) (see page 17)

[Install the Administrative Files](#) (see page 30)

[License CA ERwin Model Manager](#) (see page 31)

[Refresh the Mart After Upgrading Your License](#) (see page 32)

[Create the CA ERwin MM Mart](#) (see page 33)

Prepare for the Installation

Review the hardware and software requirements and the list of supported databases in the *Release Notes* before you install the program. You must verify that your system requirements meet at least the minimum specified requirements, and that you have verified and prepared your DBMS environment for installation.

Before you can install CA ERwin MM in your DBMS environment you must do the following:

1. Create the database on your DBMS server.
2. Install the administrative files.
3. Install or upgrade your license agreement permitting access the mart.
4. Create the mart or upgrade your current version of software when a newer version is available.

Note: You must be the database owner (dbo) for Microsoft SQL 2000 Server or Sybase, the database owner (dbo) and have the sysadmin role for Microsoft SQL 2005/2008 Server, or the database schema owner user and the DBA role in Oracle in the target database to create or delete a mart.

If you installed an earlier version of CA ERwin MM (before release 7) it is not necessary to remove that version. Because this version is installed to a new folder, you can continue to work with the earlier version, if you have sufficient disk space on your PC for both versions.

Specify Use of Foreign Characters With Microsoft SQL Server 2000

For Microsoft SQL Server 2000, it is necessary to select specific settings in the Client Network Utility to have certain foreign language characters in your models recognized.

To specify the use of foreign characters with Microsoft SQL Server 2000

1. Click Programs, Microsoft SQL Server, Client Network Utility on the Start menu.

The SQL Server Client Network Utility dialog opens.

2. Select the following check boxes on the DB-Library Options tab:

- Automatic ANSI to OEM conversion
- Use international settings

Click OK.

Your configuration is set to recognize foreign language characters in your models.

Specify Use of Foreign Characters With Microsoft SQL Server 2005

For Microsoft SQL Server 2005, it is necessary to modify your registry settings to have certain foreign language characters in your models recognized.

To specify the use of foreign characters with Microsoft SQL Server 2005

1. Click Run on the Start menu.

The Run dialog opens.

2. Enter regedit.

The Registry Editor opens.

3. Verify or Add the following registry entry:

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\Client\DB-Lib]

"AutoAnsiToOem"="ON"

"UseIntlSettings"="ON"

Click File, Exit.

Your configuration is set to recognize foreign language characters in your models.

How to Prepare Your DBMS Environment

You must perform the following setup tasks in your DBMS environment before you can install and use the software:

1. Install the DBMS.

Install the DBMS on the server where you plan to store the mart. For more information about memory and disk space requirements, see the system requirements for the supported versions of each DBMS (Microsoft SQL Server, Sybase, and Oracle).

2. Create the mart.

Use the DBMS features to create or identify the required storage objects and the mart. The specific requirements vary depending on your DBMS type.

3. Create a user profile for each CA ERwin MM user.

Use the DBMS features to create a user profile for each CA ERwin MM user in the database where CA ERwin MM is stored.

Note: Each user must have Windows authentication enabled (on databases that support Windows authentication) or a valid login name on the DBMS and permission to access the database in which the mart is stored. For more information, see your DBMS administrator's documentation.

4. Install the DBMS client software on each client workstation.

To connect a client workstation to the DBMS server, you must install the appropriate client connection software (for example, SQL*Net) on each workstation. The connection software you use depends on your DBMS type.

5. Verify the client/server connection.

Use a test command (for example, ping) to check the client/server connection between each client workstation and the DBMS server. Alternatively, use another application on a client workstation (for example, ISQL/W, SQLPLUS, or PowerBuilder) to check the connection. For more information about testing the client/server connection, see your DBMS documentation.

6. Install a client on each client workstation.

Install the client software using the CD-ROM that shipped with each client (CA ERwin Data Modeler or CA ERwin Process Modeler). Then, any user with a CA ERwin MM login and password and the appropriate DBMS client connection software can connect to the mart from that client workstation.

Tasks to Create a Microsoft SQL Server DBMS

The following tasks must be performed by the DBA and system administrator responsible for installing CA ERwin MM on a Microsoft SQL Server DBMS:

1. Use a graphical user interface (GUI) tool, ISQL (all versions), SQL Administrator, or Enterprise Manager to create the CA ERwin MM database. If a GUI tool is not available, you can use ISQL to type in the appropriate commands manually.

Your database should meet the following criteria:

- The initial size of the database file should be set to 60 MB.
- The initial size of the transaction log file should be set to 50 MB.
- Set the maximum file size to unrestricted file growth for both files (recommended, but not required).
- Increase the Set Auto grow file by 10 percent (recommended, but not required).

The new database is *owned* by the user who created it.

Set the Truncate Log on Checkpoint option and have the server generate checkpoints frequently. By selecting this option, the log is emptied periodically and should not fill up and cause rollbacks.

Note: For best performance ensure that separate devices are used to store the data and the transaction log.

2. Add users.

For users to access the mart you must either add existing logins as users to the database or create new logins and add them as users of the database. A login is an entity permitted to connect to the database server; a user gives a login the right to use a particular database. If you use Windows user names and passwords to secure database access (Windows Authentication), you must add the Windows users to your DBMS.

3. Check tempdb size.

Significant temporary space is required for installation and use. The temporary segments need at least 16 MB of available space. You should also increase available space as the number of concurrent users increase.

Microsoft SQL Server Connections

If you are using a Microsoft SQL Server as your host DBMS, CA ERwin MM uses DB_LIBRARY to connect.

Microsoft SQL Server 2005 Permissions

For SQL Server 2000, you only need the public permission assigned to save to the mart. However, when the repository is on an SQL Server 2005 instance, you must have the bulkadmin permission designated as well. The ability to do bulk inserts (which was permitted by the public permission, previously) is no longer part of the public permission. As the administrator, you must explicitly define this permission or when you attempt to save a model to a new mart created using a SQL Server 2005 database, an error "You do not have permission to use the bulk load statement." is returned.

Tasks to Create a Sybase DBMS

The following tasks must be performed by the DBA and system administrator responsible for installing the software on a Sybase database management system. Information provided in this section includes installation and Sybase connectivity software requirements.

1. Use a graphical user interface (GUI) tool, ISQL (all versions), SQL Administrator, Sybase Central Java, or Enterprise Manager to create the CA ERwin MM database. If a GUI tool is not available, you can use ISQL to type in the appropriate commands manually.

Your database should meet the following criteria:

- The size of the data device you create determines the size of the database. The minimum database size is 32 MB so there has to be at least one device that is 32 MB. You should also locate the data device on a different disk (and disk controller) than the transaction log.
- For optimum performance, ensure that separate devices are used to store the data and the transaction log. For example, you can increase performance by creating a 50 MB data device and a 25 MB log device. A minimum of 40 MB of disk space (data and log) is required.

2. Add users.

For users to access the mart you must either add existing logins as users to the database or create new logins and add them as users of the database. A login is an entity permitted to connect to the database server; a user gives a login the right to use a particular database. If you use Windows user names and passwords to secure database access (Windows Authentication), you must add the Windows users to your DBMS.

3. Check the Stored Procedure Cache.

Set the Stored Procedure Cache size to at least 8 MB. Setting it higher improves performance, especially when many users are accessing the server concurrently. Setting it lower results in fatal errors and rollbacks when the Stored Procedure Cache size is exceeded.

Note: The installation creates more than 100 stored procedures. The CA ERwin MM client invokes these stored procedures to control changes to the data in the database.

4. Check tempdb size.

Significant temporary space is required for installation and use. The temporary segments need at least 16 MB of available space. You should also increase available space as the number of concurrent users increase.

5. Check memory allocated to the database server.

Allocate at least 32 MB of RAM to the database server. The amount of RAM allocated should be half of the available RAM on the server.

Sybase Connections

If you are using Sybase as your host DBMS, CA ERwin MM uses CT_LIBRARY to connect.

Post Database Creation Tasks for Microsoft SQL and Sybase

You must run the Administrative Setup program located on the CD. The Setup program prompts you for an installation directory. By default this directory is C:\Program Files\CA\ERwin Model Manager, but the path may have been changed during installation.

Note: Do not attempt to run the Setup program unless you are certain that the computer you are using can connect to the target server. For more information about tips on how to determine if your installation computer can connect to the target server, see Microsoft SQL Server Connections or Sybase Connections.

When you install the software, the Microsoft SQL installer can be 'owner,' 'user with DB owner privileges,' or 'user with alias as DB owner privileges.' The Sybase installer can be 'owner,' 'user with sa_role,' or 'user with alias as DB owner.'

Transact SQL Commands

If a graphical DBMS access tool is not available you can use Transact SQL commands through ISQL.

- The following is an example of how to create a device using the Transact SQL DISK INIT command through ISQL:

```
DISK INIT NAME = 'mmdata',          /* The logical name. */
PHYSNAME = 'C:\SQL\DATA\mmdata.dat', /* The physical name. */
VDEVNO = 1<= virtual_device_number => 255
/* System dependent. */
SIZE = number_of_2K_blocks          /* 1024 here is 2MB!!! */
[, VSTART = virtual_address,        /* Optional */
CNTRLTYPE = controller_number]     /* Optional */
```

- The following is an example of how to create a database using the Transact SQL CREATE DATABASE command through ISQL:

```
CREATE DATABASE mmmaster
[ON {DEFAULT | database_device} [= size_in_megabytes] /* The device created in #1. */
[, database_device [= size_in_megabytes]]...] /* A database can span devices. */
[LOG ON database_device [= size_in_megabytes>] /* Separate log device. */
[, database_device [= size_in_megabytes]]...] /* A transaction log can span devices. */
```

- The following is an example of how to add logins to the database with the sp_addlogin and sp_adduser commands using Transact SQL through ISQL:

```
sp_addlogin login_id [, passwd [, defdb [, deflanguage]]]
```

and

```
sp_adduser login_id [, username [, grpname]]
```

Once you have executed these commands, the DBA can alias an existing login as the Database Owner (dbo) or change the dbo to an existing login using sp_changedbowner. Use ISQL to execute the following:

```
sp_changedbowner login_id [,true]
```

Tasks to Create an Oracle DBMS

The following tasks must be performed by the DBA and system administrator responsible for installing the software on an Oracle database management system. Information provided in this section includes installation and Oracle connectivity software requirements.

These steps can be performed using graphical tools but you can also use SQL*Plus (all versions), SQL*DBA (either command line or full screen mode), or the Oracle Enterprise Management Console that started shipping with Oracle Version 8.0. Examples of SQL commands are included where appropriate. Data file paths, data file sizes, role names, and user names are included for example only.

1. Check SYSTEM tablespace.

The installation creates several stored procedures. All triggers, stored procedures, and packages are kept in the Oracle SYSTEM tablespace. The standard size of the SYSTEM tablespace assumes that you are not using procedural options, so the SYSTEM tablespace often needs to be expanded. If other Oracle applications are not using procedural code, then the SYSTEM tablespace should be expanded to 32 MB. If other Oracle applications also use procedural code, expand the SYSTEM tablespace to at least 32 MB.

2. Check Rollback Segment tablespace.

If your instance uses UNDO tablespace, do not create rollback segments.

Significant rollback space is required for installation and use. The rollback segments should be in their own separate tablespace and each have at least 16 MB of available space. There should be one rollback segment for every four concurrent users, with a maximum of 50 rollback segments. The available space should scale upward with increasing numbers of rollback segments. Finally, the rollback segment optimal parameter should be set to control rollback segment growth and space consumption.

Note: For Steps 3, 4, and 5, use Dictionary-managed tablespaces.

3. Create a data tablespace of at least 32 MB.

For example:

```
CREATE TABLESPACE MyMart
```

```
DATAFILE '/db01/oracle/rdbms9i/data/mymart.ora' SIZE 100M;
```

Or For Locally Managed extents:

```
CREATE TABLESPACE Mymart
```

```
DATAFILE '/db01/oracle/rdbms9i/data/mymart.ora' SIZE 100M
```

```
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;
```

4. Create a index tablespace of at least 32 MB.

For example:

```
CREATE TABLESPACE MMARTINDEX  
  
DATAFILE '/db02/oracle/rdbms9i/data/mmartindex.ora' SIZE 75M;  
  
Or For Locally Managed extents:  
  
CREATE TABLESPACE MMARTINDEX  
  
DATAFILE '/db02/oracle/rdbms9i/data/mmartindex.ora' SIZE 75M  
  
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;
```

5. Create a temporary tablespace.

For example:

```
CREATE TEMPORARY TABLESPACE MMTEMP TEMPFILE '/db03/oracle/rdbms9i/data/mmarttemp.ora'  
SIZE 50M;  
  
Or For Locally Managed extents:  
  
CREATE TEMPORARY TABLESPACE MMTEMP TEMPFILE '/db03/oracle/rdbms9i/data/mmarttemp.ora'  
SIZE 50M  
  
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1M;
```

Note: For more details on syntax and options regarding tablespace creation, see the appropriate Oracle documentation.

6. Create an Oracle user with DBA privileges to be used by the CA ERwin MM Installer or designated schema owner.

Assign the data tablespace as this user's default tablespace, and the temporary tablespace as this user's temporary tablespace.

For example:

```
CREATE USER STEVE IDENTIFIED BY STEVE  
  
DEFAULT TABLESPACE MyMart  
  
TEMPORARY TABLESPACE MMTEMP  
  
QUOTA UNLIMITED ON MyMart  
  
QUOTA UNLIMITED ON MMARTINDEX;
```

7. Create the CA ERwin MM Installer role.

The following example is the role required by the Oracle user installing CA ERwin MM.

```
CREATE ROLE MMINSTALL;
```

8. Grant Oracle privileges to the Installer role.

The following example shows the Oracle privileges that the CA ERwin MM Installer needs to install on Oracle. For the last command, you must log in as sys with the sysdba role in the user-name or the command will fail.

```
grant create sequence to MMINSTALL;  
grant create table to MMINSTALL;  
grant create view to MMINSTALL;  
grant drop public synonym to MMINSTALL;  
grant create public synonym to MMINSTALL;  
grant create procedure to MMINSTALL;  
grant select on dba_data_files to MMINSTALL;
```

9. Create the CA ERwin MM User role.

For example:

```
CREATE ROLE MMUSER;
```

When you select this role as the CA ERwin MM User role during Step 4 of the installation procedure, the Setup program generates grant statements that grant object level privileges to this role.

10. Grant the create session Oracle privilege to the User role.

For example:

```
grant create session to MMUSER;
```

Note: The create session privilege is the only privilege that an Oracle user needs to use the database.

11. Grant the CA ERwin MM User role to each Oracle user that uses the database.

For example:

```
grant MMUSER to USER1;  
grant MMUSER to USER2;  
grant MMUSER to USER3;
```

Important! You must specify the User role name created in Step 9 for these grant statements.

12. Grant the CA ERwin MM User role to the CA ERwin MM Installer role.

For example:

```
grant MMUSER to MMINSTALL;
```


13. Grant the CA ERwin MM Installer role to the CA ERwin MM Installer user. You must also grant DBA privileges to the user you designate as the CA ERwin MM Installer.

For example:

```
grant MMINSTALL to STEVE;
```

```
grant DBA to STEVE;
```

Note: The DBA role is necessary only during the installation or upgrade. You can strip the installer of the DBA role after the installation or upgrade is complete.

14. Run the Setup program.

When the setup is complete, select the Initialize CA ERwin MM option and connect to Oracle as the CA ERwin MM Installer user (for example, STEVE).

15. Select the CA ERwin MM tablespaces and CA ERwin MM User role.

When prompted for tablespace and role information, select the CA ERwin MM data tablespace, the CA ERwin MM index tablespace, and the CA ERwin MM User role (the role created in Step 9).

16. Add the CA ERwin MM Installer user as the CA ERwin MM administrator.

When the Security Manager is opened, your user name is automatically added to the Administrator profile in the Security Profile list. You can also designate one of the users with the MMUSER role (from step 12) for routine operations. This completes the setup and you can begin using the software.

Note: After successful installation, you can revoke the CA ERwin MM Installer role and grant the CA ERwin MM User role to the CA ERwin MM Installer user.

Database Objects Installation

Install the software on an Oracle DBMS to create the following database objects:

- Tables
- Indexes
- Stored procedures
- Public synonyms

Oracle Connections

Oracle SQL*Net is used for client access to the database server. Although Oracle provides two utilities (SQL*Net Configuration tool and Network Manager tool) for defining your network, clients, and server, you need a network administrator to perform this task. The server system administrator is usually the best candidate.

Note: For more information about these utilities, see the Oracle *SQL*Net Administrator's Guide*.

SQL*Net requires certain files on both the server and client. These files can be generated by the Oracle networking tools, but can also be created using a text editor. Because Oracle does not support those sites that generate these files manually, it is better to use the appropriate Oracle utilities. The following lists the files required by SQL*Net:

File Name	Required On	File Contents
TNSNAMES.ORA	Client and Server	A list of service names and connect descriptors for network destinations (tells the client where it can make connections)
SQLNET.ORA	Client and Server	A list of optional diagnostic parameters
LISTENER.ORA	Server only	A list of names and addresses of all listeners on a computer and the Oracle SIDs for the databases known on that computer

Oracle SQL Commands

If a graphical DBMS access tool is not available, you can use Oracle SQL commands through SQL*DBA or SQL*Plus.

- The following is an example of how to create a tablespace using the CREATE TABLESPACE command:

```
CREATE TABLESPACE mm_data /* The tablespace name. */
DATAFILE 'C:\ORANT\DATABASE\mmdata.dat' /* The data file name. */
SIZE integer_value K or M or G /* The data file size */
DEFAULT STORAGE ( /* The default storage parameters */
  INITIAL integer_value K or M or G /* The initial extent size */
  NEXT integer_value K or M or G /* The next extent size */
  PCTINCREASE integer_value /* The percent to grow extents */
  MINEXTENTS integer_value /* The minimum number of extents */
  MAXEXTENTS integer_value /* The maximum number of extents */
);
```

- The following is an example of how to create an Oracle user using the CREATE USER command:

```
CREATE USER mm_user_1 /* The user id name */
IDENTIFIED BY password /* The user password */
DEAFULT TABLESPACE tablespace_name /* The user's default tablespace */
TEMPORARY TABLESPACE tablespace_name /* The user's temporary tablespace */
QUOTA unlimited_or_integer_K_M_G ON tablespace_name /* The user's quota on a tablespace */
;
```

- The following is an example of how to grant user privileges using the GRANT command:

```
GRANT role_or_privilege_name
TO user_or_role
;
```

CA ERwin MM Mart Creation Requirements

To create the CA ERwin MM mart, you must meet one of the following requirements:

- you must be the database owner (dbo) in the target database on the Microsoft SQL 2000 Server or Sybase server

Note: The database owner (dbo) is necessary only during the software installation or upgrade. It is not necessary after you create the mart.

- you must be the database owner (dbo) and have the sysadmin role in the target database on the Microsoft SQL 2005/2008 Server

Note: The sysadmin role for the database owner (dbo) is necessary on the Microsoft SQL 2005/2008 Server for CA ERwin MM functionality to work properly (during and after the Model Manager create).

- you must be the database schema owner user and have the DBA role in the target database on the Oracle server

Server Connection Software

The software you use to connect your client workstations with your server depends on the type of DBMS you are using and your personal preference. The following lists some commonly used DBMS client software connection packages:

DBMS	Suggested Software
Microsoft SQL Server or Sybase	Microsoft SQL Server client installation, Open Client/C Developer's Kit for PC Windows and Net-Library for PC Windows, CT-LIB, or equivalent software
Oracle	Oracle "Required Files," SQL*Net, and TCP/IP or equivalent communication software

Note: For more information about memory and disk space requirements, see your DBMS documentation.

Connection Prerequisites

You connect to CA ERwin MM on the server through the client software contained in CA ERwin Data Modeler, CA ERwin Process Modeler, or CA ERwin Model Navigator. You must perform the following tasks before you can connect to the mart through the client:

- The administrator of the relational database management system (DBMS) must give users permission to access the DBMS, on which the CA ERwin MM database is set up.
- The CA ERwin MM administrator must install the administrative files on the server, and set up the database. The administrator also defines user security permissions and sets up the library structure in which models are organized. When you install the CA ERwin MM software, the administrator is automatically granted Administrator security status. This lets the administrator assign user permissions and perform all required maintenance or administrative tasks.
- At least one of the client programs must be installed on the client workstation. System requirements for the client workstations are the same as those for the CA ERwin MM client software. You can find the requirements in the documentation provided with these products.
- The client connectivity software for your DBMS on the server must be installed on the client workstation so that you can connect to the DBMS on which the database resides.

Install the Administrative Files

The Installation Wizard makes it easy for you to install, change, or remove the software. The installation of the administrative files is the same for each DBMS supported (Microsoft SQL Server, Sybase, or Oracle).

Note: You must be the database owner (dbo) for Microsoft SQL 2000 Server or Sybase, the database owner (dbo) and have the sysadmin role for Microsoft SQL 2005/2008 Server, or the database schema owner user and the DBA role in Oracle in the target database to create or delete a mart.

To install the administrative files

1. Insert the installation DVD or double-click the file you downloaded from the online CA product page.

The Installation Wizard opens.

2. Follow the prompts in the wizard to proceed. If you install from the DVD, you are prompted to select the applications you wish to change or install. Click Next.

A License Agreement opens for you to review.

3. Select the option to accept the terms as described in the License Agreement. Click Next.

The wizard asks a series of questions you must answer, including:

- Customer Information
- Destination folder

4. On the Custom Setup Type screen, review the program features that will be installed. You can also evaluate the space needed to install each feature.

Click Next to continue and then click Install to start the installation.

The program files are copied.

Click Finish to exit the wizard.

The installation is complete and the Connection Manager opens to initialize your mart.

Important! You must install or upgrade the license key before you initialize the mart. If you have not already licensed CA ERwin MM, you can license the program when the install process is complete. Follow the on-screen prompts to open the License Verification dialog and enter the license key provided with the product CD.

License CA ERwin Model Manager

If you have not already licensed CA ERwin MM, you can license the program at this time. Locate the 25-character License Key that accompanies the product CD. Typically, this key is found on the back of the CD sleeve. You need this License Key to complete the license process.

To license CA ERwin MM

1. Click Programs, CA, ERwin, ERwin Model Manager r7.3, Add MM License on the Start menu.

The License verification dialog opens.

2. Select a component in the Component List in the License Verification dialog and select an option:

Use Trial Version

Specifies to use the Try Buy license. The product runs for 30 days before requiring a license.

Use License Key

Specifies to enter a license key.

3. Enter the License Key in the License Key field for the first component listed in the License Verification dialog, and click Add.

The information you entered displays in the License Keys field.

Note: If you enter an Upgrade license key in the License Key field, and the component is not currently licensed on your system, the Registration ID field appears, permitting you to enter the Registration ID for your upgraded component.

4. Select the next component you are licensing from the Component List, enter the License Key for this component in the License Key fields and click Add.

Repeat these steps until you have entered License Keys for all of the components you are licensing.

5. Click Continue.

The installation process continues.

Refresh the Mart After Upgrading Your License

After you upgrade the license file, you must refresh the mart.

To refresh the mart

1. Click Programs, CA, ERwin, ERwin Model Manager r7.3, Model Manager on the Start menu.

The Connection Manager dialog opens.

2. Connect to the database where you upgraded the license.

The connection is made.

3. Click Close.

The database refresh is automatic.

Note: If you upgrade your user license with the software upgrade, the Setup program automatically detects your new license key and a message displays confirming the license upgrade. You should install the new license file before you install the software upgrade.

Create the CA ERwin MM Mart

After you create the CA ERwin MM database on your DBMS and install the administrative files, you must create the CA ERwin MM mart to create the tables, install the stored procedures that manage client workgroup models in the mart, and prepare the mart for use with CA ERwin Data Modeler and CA ERwin Process Modeler. You must be the database owner (dbo) for Microsoft SQL 2000 Server or Sybase, the database owner (dbo) and have the sysadmin role for Microsoft SQL 2005/2008 Server, or the database schema owner user and the DBA role in Oracle in the target database to create or delete a mart.


Note: The CA ERwin MM mart and the CA ERwin MM Control Tables (m7Master and m7License) must be installed on the same database for Microsoft SQL Server or Sybase DBMS.

Important! You must install or upgrade the license key before you create the CA ERwin MM mart.

To create the mart

1. Log on to your DBMS that contains CA ERwin MM as the dbo or schema owner. Click Programs, CA, ERwin, ERwin Model Manager r7.3, Model Manager on the Start menu.

The Connection Manager opens.



The dialog box is titled "ERwin Model Manager Connection Manager". It contains the following fields and controls:

- Database:** A dropdown menu showing "SQL Server 2000 (using db-lib)".
- Authentication:** A dropdown menu showing "Windows Authentication".
- User Name:** A text box containing "\Server\UserName".
- Password:** An empty text box.
- Parameters Table:** A table with two columns: "Parameters" and "Value".

Parameters	Value
Server:	MM_Server_2000
Database:	Test_Mart
- Recent Connections:** A list box with five empty rows and a scrollbar.
- Buttons:** "Connect", "Disconnect", "Close", and "Help".

2. Complete the following information:

Database

Identifies the type of relational database management system (DBMS) you will connect to. Select from the current list of supported databases.

Authentication

Windows Authentication

Specifies the use of Windows user names and passwords to secure database access. You must be logged on to your computer as the user who is the dbo of the database. (Available only for Oracle and SQL Server.)

Database Authentication

Specifies the use of a local user name and password for the connection.

Parameters/Value Options

Connection Type (Microsoft SQL Server 2005 Only)

Specifies the use of Native Connection to connect using the API provided by the SQL Server Native client software or ODBC data to connect using the ODBC data source you have defined.

Server

Identifies the server name.

Database

Identifies the name of the CA ERwin MM database or mart.

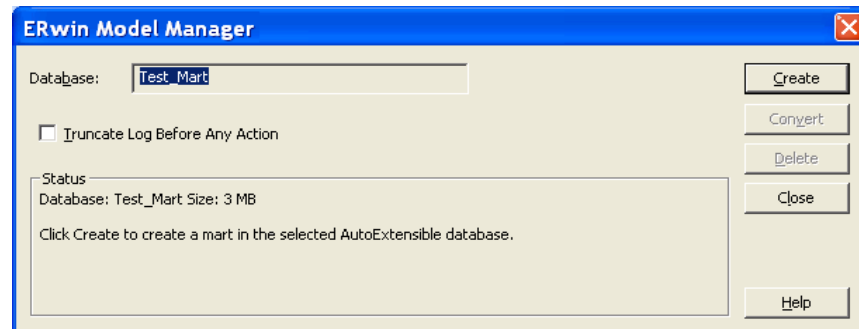
Connection String (Oracle Only)

Specifies the connection string (TNSNames entry).

You can select a database connection from the Recent Connections panel to automatically populate the Database or Connection String previously used.

Click Connect.

The ERwin Model Manager dialog opens.



3. Select from the following options:

Database (SQL and Sybase DBMS)

Identifies the master database name to create the tables and stored procedures.

CA ERwin MM Role (Oracle DBMS)

Identifies the Oracle MMUSER security role. This role is created in the procedure for creating the CA ERwin MM database on Oracle. Other user roles can connect to CA ERwin MM, but they will encounter access problems.

Table Tablespace

Indicates where you want to store the CA ERwin MM tables.

Note: The drop-down box is not active if there is only one tablespace.

Index Tablespace

Indicates where you want to store the CA ERwin MM indexes.

Note: You can store tables and indexes in the same tablespace or in different tablespaces.

Truncate Log Before Any Action

Specifies to remove the database transaction log. Truncating the database transaction log is especially important for Microsoft SQL Server and Sybase users since a large transaction log can cause the database to hang.

Note: When you initialize a Microsoft SQL Server 2008 database, it is not necessary to select the "Truncate Log Before Any Action" option. If you select this check box, and click Create, the following error message displays: "Incorrect syntax near the keyword 'TRANSACTION'." To avoid this message, clear the check box for "Truncate Log Before Any Action" before you click Create.

Click Create.

The CA ERwin MM objects are created and the Security Manager dialog opens.

Important! If you do not see the Security Manager dialog, it can be hidden behind other open applications. Minimize all applications or use the Alt-Tab keys until you find it.

Your user name is automatically added to the Administrator profile in the Security Profile list.

4. Use the controls in the Security Manager dialog to assign additional security profiles. Click OK when complete.

Note: You can also do this later from CA ERwin Data Modeler or CA ERwin Process Modeler.

The Setup Complete message opens.



5. Click OK.

The mart is created.

Chapter 3: Upgrade

This section contains the following topics:

[Software Upgrades](#) (see page 39)

[Conversion Process](#) (see page 39)

[How to Upgrade the Mart](#) (see page 42)

[Install the Software Upgrade](#) (see page 44)

[Convert or Update a CA ERwin MM Mart](#) (see page 46)

Software Upgrades

New versions of the software and supported client software products are periodically released. The upgrade process prepares the tables and stored procedures to accommodate new functionality and client functionality, but does not change or delete any data stored in the mart. To upgrade the software, you must be assigned to the Administrator security profile and logged on to the DBMS.

You must be the database owner (dbo) for Microsoft SQL 2000 Server or Sybase, the database owner (dbo) and have the sysadmin role for Microsoft SQL 2005/2008 Server, or the database schema owner user and the DBA role in Oracle in the target database to create or delete a mart.

The software should be installed or upgraded on the computer that houses the CA License Key. This workstation is typically the same computer that the administrator uses or the first computer that was used to install the software.

Note: For some software upgrades, a new license key is required.

The mart upgrade process is slightly different depending on your DBMS. For more information about specific procedures, see the particular section on the database you are running.

Conversion Process

The conversion systematically converts a mart, including all CA ERwin MM properties and all client models.

The time it takes to convert a mart varies for each installation. Server speed, size, complexity, and the number of models in the mart are several factors that influence conversion time.

Property Conversion

The conversion process upgrades the following CA ERwin MM properties:

- All user names and security profiles (even if the user no longer exists in the database)
- All session information
- All libraries including the preservation of the complete library structure

Access information is not converted. Access information is stored internally for tracking purposes.

Model Property Conversion

The model conversion process converts all CA ERwin Data Modeler and CA ERwin Process Modeler model properties including the following:

Model data

CA ERwin Data Modeler and CA ERwin Process Modeler models in the mart are converted. The converted models are the same as if opening an .er1 file with the upgraded client and saving the model to CA ERwin MM.

All model objects

CA ERwin Data Modeler and CA ERwin Process Modeler accept all model objects from earlier versions.

All object properties

All model properties such as object names and user-defined properties (UDPs) are fully retained.

Conversion Methods

The following conversion methods are available to convert your existing client models:

Manual Conversion

Moves your models from the old mart to the new mart, one model at a time.

Automated Conversion

Installs the new software version and initializes the mart in the same location as the old mart. Click Convert to automatically convert all existing models in the old mart to the new mart.

Manual Conversion

Manual conversion means moving your models from your old mart to your new mart, one model at a time. The following are some advantages to the manual method:

- Selectively choose the models you want to move, and leave those that you do not.
- “Clean up” your mart if there are many models or versions which are no longer in use, and which would extend conversion time.
- Run both CA ERwin MM versions simultaneously, and move incrementally, model by model, to the new mart.
- Host your new mart on a different server or database platform than your old mart version.

The disadvantages are that the version history is lost on models checked in to the new mart. Being a manual process, it is time consuming and labor intensive.

Automated Conversion

The automated conversion uses the Update option on the CA ERwin Model Manager dialog. Your existing CA ERwin Data Modeler and CA ERwin Process Modeler models in the old mart are converted to the new mart. All models are converted automatically at the same time.

The main advantage to this method is that it is automated and all models retain their full history. The disadvantage is that it takes a long time if your mart is large, or your client/server hardware is limited. The automated conversion must be done in the same database instance as your previous version. This may require more space than is available, or require you to move your old version of the mart, if you cannot create additional space.

How to Upgrade the Mart

It is important to run through the upgrade process in a test environment before attempting to upgrade your production environment. By doing so, you become aware of any challenges that exist, and have fewer surprises when you actually convert your production environment. To convert your mart as efficiently and as fast as possible, follow these recommendations:

- Make a full backup of the current database using the backup utility for your DBMS before upgrading.
- Open and save all models in your mart using the current version of the client before you convert to CA ERwin MM r7.3.
- Ensure that all users check in any off-line models and ensure that there are no active sessions on the mart and that no models are checked out.
- Remove obsolete models in the current database to reduce the size, which improves the conversion speed.
- Verify that CA ERwin Data Modeler and CA ERwin Process Modeler models are synchronized if you synchronize data.
- Verify that your client version is compatible with the database version to which you are migrating.
- Increase the size of the database to a minimum of three times the size of the previous database for temporary conversion objects.
- Verify there are no outstanding technical support issues related to CA ERwin MM.
- Turn on AUTOEXTEND to automatically extend the CA ERwin MM data file when full (Oracle DBMSs only).
- Make sure the Auto grow file is turned on, with your choice of File Growth and unrestricted File Growth for both data files and transaction logs (SQL2000 DBMSs only).
- Review the Release Notes file.
- Run the upgrade during off-peak hours.
- Truncate the database log file.
- Start the conversion from the server to eliminate the possibility of network speed becoming a bottleneck. Have at least 1 GB on the client, typically more memory means faster performance.
- Log on to your DBMS as either the dbo (Microsoft SQL or Sybase) or schema owner (Oracle) of the database that contains CA ERwin MM.

Microsoft SQL Server or Sybase DBMS Upgrade Considerations

On a Microsoft SQL Server or Sybase DBMS, if you are updating from a previous CA ERwin MM version and the system logs are full, you may not be able to upgrade your previous CA ERwin MM installation with the current version. Before you install the new version, use a DBMS utility (such as Microsoft SQL Object Manager) and type the appropriate command to purge the transaction log.

Oracle DBMS Upgrade Considerations

On an Oracle DBMS, before you start the conversion, it is important that you turn on AUTOEXTEND to automatically extend the CA ERwin MM data file when full.

CA ERwin MM 4.1.4 to CA ERwin MM r7.3 Conversion

Verify that the mart is Version 4.1.4 (Build 3643 or later). If it is a version earlier than 4.1.4, you must upgrade to 4.1.4 before you can upgrade to r7.3. We recommend you upgrade to the latest Service Pack for CA ERwin MM Version 4.1.4. You must open and save your models older than CA ERwin MM Version 4.1.4 in CA ERwin MM Version 4.1.4 before you can upgrade to CA ERwin MM r7.3.

See <http://ca.com/supportconnect/> for details on upgrading to CA ERwin MM Version 4.1.4.

Install the Software Upgrade

To upgrade the software to a new version you must log on to your DBMS. You must be the database owner (dbo) for Microsoft SQL 2000 Server or Sybase, the database owner (dbo) and have the sysadmin role for Microsoft SQL 2005/2008 Server, or the database schema owner user and the DBA role in Oracle in the target database to create or delete a mart.

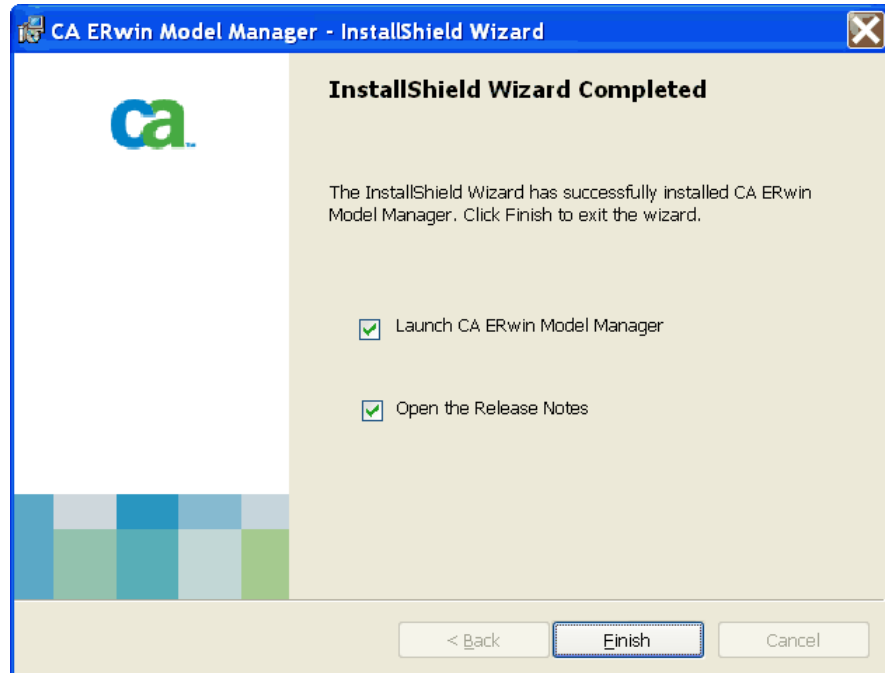
Note: For some software upgrades, a new license key is required. Locate the 25-character License Key that accompanies the product CD. Typically, this key is found on the back of the CD sleeve.

To install the software upgrade

1. Close all client connections and insert the CD in your CD-ROM drive.
If Autorun is enabled, you are prompted to install. If not, double-click Setup.exe from the CD drive.
The Welcome dialog opens.
2. Click Next to continue.
A License Agreement opens for you to review.
Select the option to accept the terms as described in the License Agreement and continue. Follow the instructions provided by the installation wizard.

3. When the install completes, a final screen displays with the following two options:

- Launch CA ERwin Model Manager
- Open the Release Notes



Both check boxes are selected by default. If you want to update your mart at a later time, clear the Launch CA ERwin Model Manager check box. When you are ready to update the mart, you can launch the product from the Start menu.

Click Finish to exit the wizard.

The software upgrade installation completes and the Connection Manager opens to convert or update the mart.

Convert or Update a CA ERwin MM Mart

All the existing CA ERwin Data Modeler and CA ERwin Process Modeler models in the mart are initialized and prepared for the r7.3 internal structure. The model is not fully converted or updated until you open and save the model in CA ERwin MM r7.3 with CA ERwin Data Modeler or CA ERwin Process Modeler for the first time. It can take a few minutes longer to open the model the first time to complete the process.

To convert a mart

1. Log on to your DBMS that contains CA ERwin MM. You must be the database owner (dbo) for Microsoft SQL 2000 Server or Sybase, the database owner (dbo) and have the sysadmin role for Microsoft SQL 2005/2008 Server, or the database schema owner user and the DBA role in Oracle in the target database to create or delete a mart. Click Programs, CA, ERwin, ERwin Model Manager r7.3, Model Manager on the Start menu.

The Connection Manager opens.

Parameters	Value
Server:	MM_Server_2000
Database:	Test_Mart

2. Complete the following information:

Database

Identifies the type of relational database management system (DBMS) you will connect to. Select from the current list of supported databases.

Authentication

Windows Authentication

Specifies the use of Windows user names and passwords to secure database access. You must be logged on to your computer as the user who is the dbo of the database. (Available only for Oracle and SQL Server.)

Database Authentication

Specifies the use of a local user name and password for the connection.

Parameters/Value Options

Connection Type (Microsoft SQL Server 2005 Only)

Specifies the use of Native Connection to connect using the API provided by the SQL Server Native client software or ODBC data to connect using the ODBC data source you have defined.

Server

Identifies the server name.

Database

Identifies the name of the CA ERwin MM database or mart.

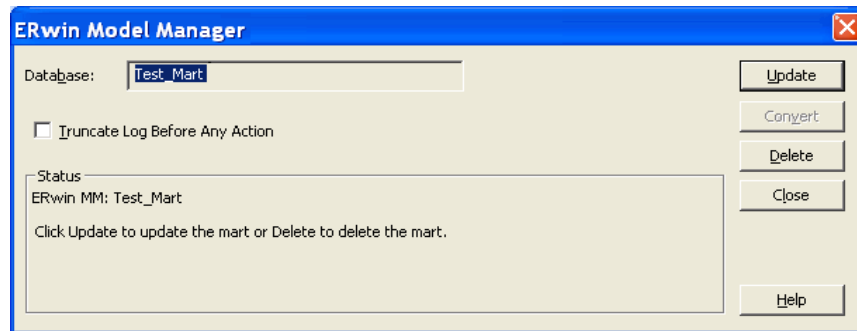
Connection String (Oracle Only)

Specifies the connection string (TNSNames entry).

You can select a database connection from the Recent Connections panel to automatically populate the Database or Connection String previously used.

Click Connect.

The ERwin Model Manager dialog opens.



3. Select from the following options:

Database (SQL and Sybase DBMS)

Identifies the master database name to create the tables and stored procedures.

CA ERwin MM Role (Oracle DBMS)

Identifies the Oracle MMUSER security role. This role is created in the procedure for creating the CA ERwin MM database on Oracle. Other user roles can connect to CA ERwin MM, but they will encounter access problems.

Table Tablespace

Indicates where you want to store the CA ERwin MM tables.

Note: The drop-down box is not active if there is only one tablespace.

Index Tablespace

Indicates where you want to store the CA ERwin MM indexes.

Note: You can store tables and indexes in the same tablespace or in different tablespaces.

Truncate Log Before Any Action

Specifies to remove the database transaction log. Truncating the database transaction log is especially important for Microsoft SQL Server and Sybase users since a large transaction log can cause the database to hang.

Note: When you initialize a Microsoft SQL Server 2008 database, it is not necessary to select the "Truncate Log Before Any Action" option. If you select this check box, and click Create, the following error message displays: "Incorrect syntax near the keyword 'TRANSACTION'." To avoid this message, clear the check box for "Truncate Log Before Any Action" before you click Create.

Select one of the following after verifying the mart name in the Status window:

Create

Creates an empty mart. You can manually open and save existing CA ERwin Data Modeler models to this new mart.

Update

Creates a new mart with the current version and updates the data from the old mart to the new mart. The Update button is enabled only when an earlier version of a CA ERwin MM is detected. When you update your mart, the old mart is no longer usable in the previous version.

Note: The Create and Update buttons are mutually exclusive and the text of the button changes depending if a previous version of the mart is detected. If the Create button is not enabled, then a previous version of a mart is detected.

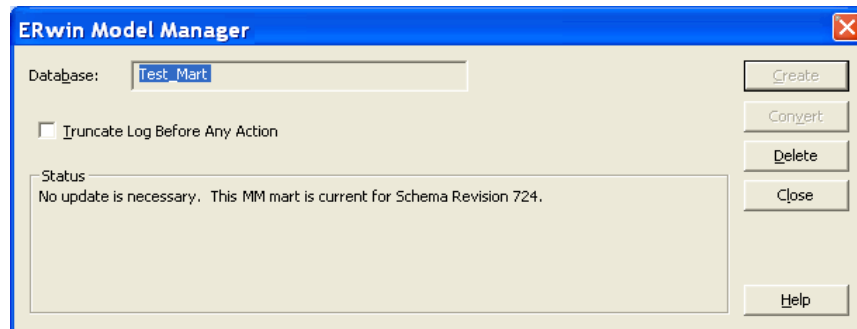
Convert

Creates a new mart with the current version and copies the data from the old mart to the new mart. The old mart is retained in the same database.

The dialog displays conversion progress messages.

- Click OK on the Upgrade has Successfully Completed message dialog.

The ERwin Model Manager dialog displays the status.



- Click Close to exit out of the ERwin Model Manager dialog.

The administrative setup is complete.

- Click OK on the Setup Complete dialog.

The conversion is complete.

The same user profile assignments exist as before the conversion. If you must make changes to user profiles, you can access the Security Manager on the Services menu or click the User Permissions icon on the CA ERwin MM Toolbar.

Appendix A: Microsoft SQL and Sybase Server Tuning Recommendations

This section contains the following topics:

[Physical Tuning Parameters](#) (see page 51)

[Server-Level Tuning Parameters](#) (see page 52)

[Named Cache Configuration](#) (see page 53)

[Database Tuning Parameters](#) (see page 53)

[Database and Log Sizing](#) (see page 55)

[How to Maintain an Efficient Database](#) (see page 55)

Physical Tuning Parameters

The mart is dynamic, with many queries and data manipulations performed with each model save and load. Although the Microsoft SQL Server and Sybase architectures differ in some ways, the basic operation and configuration of both server environments is the same.

The configuration changes that must be made to Microsoft SQL Server and Sybase for optimum performance with maximum fault tolerance and recoverability are provided.

Hardware Configurations

Consider implementing CA ERwin MM on a multiprocessor system, especially if multiple data modelers are saving information simultaneously. Install the software on a stand-alone Microsoft SQL Server, if possible, or on a development computer. The database overhead noticeably affects other databases running on the same system during save and load operations. If few modelers save and load simultaneously, the overhead is less.

To improve caching performance, implement CA ERwin MM on servers that have at least 512 MB of RAM.

Database Placement

CA ERwin MM databases are written to in bursts of activity. Stored procedures and bulk insert statements are used to minimize overhead when saving data to Microsoft SQL Server. To optimize performance, place the database on low activity Microsoft SQL Server files. If possible, place the log and data segments on different physical drives to reduce contention between log writes and database reads.

Data Redundancy

Make sure that the transaction log is mirrored on a separate drive to maximize recoverability of the database. If the up-time of the database is considered critical, mirror the database and the transaction logs; you can also mirror the other Microsoft SQL Server files. Consider OS mirroring to speed the mirroring process and make it easier to manage.

Server-Level Tuning Parameters

You should perform the following server-level tuning tasks:

- Configure Microsoft SQL Server to use as much memory as possible. Allocating more memory to the Microsoft SQL Server caching mechanisms means less physical reads from the disk and improved database query performance.
- Allow several megabytes of disk space for the procedure cache because CA ERwin MM uses many stored procedures.
- Configure two additional user connections for each simultaneous CA ERwin MM user. Two connections per session are used.

The following shows the recommended configuration parameters for a Microsoft SQL Server running CA ERwin MM:

Parameter Type	Recommended Value	Notes
memory (Sybase 11: "total memory")	512 MB minimum Note: 1 GB is recommended	More memory implies less physical I/O
procedural cache	30 percent	CA ERwin MM is stored procedure intensive
user connections	2 times the number of concurrent sessions	Add this value to existing user connection values

Named Cache Configuration

For Sybase systems, consider setting up a 4 KB pool for the default data cache. Sybase writes I/O to the log more often in 4 KB increments than in 2 KB increments (the default). You can set up a 4 KB I/O pool using `sp_poolconfig`, but you must restart the server to enable the 4 KB I/O writes to the log. Set the pool up as a smaller subset of the cache.

Note: Be careful when making changes to the data caching systems in Sybase. Monitor your changes with SQL Monitor or `sp_sysmon` to ensure that the changes you make do not starve the 2 KB I/O pool or other caches.

The following shows the recommended size of the 4 KB I/O pool for small, medium, and large servers:

Server Size	Data Cache Size	4 KB I/O Pool Size
Small	128 MB	4 MB
Medium	512 MB	6 MB
Large	1 GB	10 MB

Database Tuning Parameters

You can make database tuning adjustments in three different areas:

- Transaction log
- Threshold procedures
- Database options

Transaction Log

The transaction log keeps a before and after image of each change made in the database, and can be backed up independently of the database by issuing a `DUMP TRANSACTION` command. Microsoft SQL Server keeps a transaction log for each database.

Note: A side effect of the `DUMP TRANSACTION` command is that it clears the inactive transactions from the log to reduce its size.

Back up the transaction log frequently. This keeps the transaction log small and reduces the amount of data lost in the event of a severe database corruption.

Threshold Procedures

For Sybase, enable a threshold procedure for the last-chance threshold to back up the log when it runs out of disk space.

Note: Because Microsoft SQL Server does not provide such an option, you must back up the log frequently to keep the log small.

The following shows a sample Sybase threshold action procedure:

```
create procedure sp_thresholdaction
@dbname char(40),
@segment_name char(40),
@space_left int,
@status int
as
/* make the thresholdaction procedure backup the log */
declare @backdevice varchar(255)
select @backdevice= '/u/backups/tranfile'+
convert(char(8),getdate(),4)
dump transaction mart to @backdevice
go
```

"trunc. log on chkpt" Option

To ensure maximum recoverability, do not enable the "trunc. log on chkpt" option, since the log is cleared automatically after each checkpoint operation. If the database device becomes damaged with the log device still active and this option is set, the database cannot be recovered using the log files since they are almost empty.

Database and Log Sizing

Since the database is a dynamic environment, allow plenty of space for the models you create. Tests have shown that models generate between 10 KB and 20 KB of data per entity during initial save times. Log overhead per object is approximately 15 KB to 21 KB per entity. Allocate 50 to 65 percent of the database size to the log to avoid running out of room in the log for a typical model save.

An average mart can range in size between 200 MB and 1 Gig. Size the transaction log accordingly. A 50 MB data device can store several large models (about 2,500 entities and 100,000 total objects), but keep in mind that the database becomes more flexible and has less storage space issues when the data device is large.

The following table shows the recommended database size and log size for small, medium, and large marts:

Database Size	Database Device	Transaction Log Device
Small	200 MB	100 MB
Medium	500 MB	200 MB
Large	1 GB	400 MB

How to Maintain an Efficient Database

To maintain an efficient database, perform the following maintenance tasks:

- Run `UPDATE STATISTICS` and execute `sp_recompile` frequently on every table in the database. This keeps the statistics up-to-date for the indexes, resulting in better overall performance.
- Periodically recreate the clustered indexes in the database to reduce fragmentation. Be sure to back up the database and transaction log daily.
- Run `DBCC CheckDB()`, `DBCC CheckCatalog()`, and `DBCC CheckAlloc()` or `DBCC NewAlloc()` on the database nightly, to check for corruption and inconsistencies in the database. Check the output of these queries and look for keywords like corrupt. Any problems detected by these commands are sent to the query output, so save the files and scan them regularly.

The following table shows maintenance tasks and the recommended frequency for performing these tasks:

Maintenance Task	Recommended Frequency	Reason
DBCC Checkdb	Nightly	Check for corruption in databases.
DBCC NewAlloc	Weekly	Check for allocation corruption (Microsoft SQL Server only).
DBCC CheckAlloc	Weekly	Check for allocation corruption (Microsoft SQL Server only).
DBCC CheckCatalog	Weekly	Check for system table inconsistencies.
UPDATE STATISTICS	Nightly	Recreates the statistics page for each index.
EXEC sp_recompile	Nightly	Tells which stored procedures have changed.
Backup Database	Weekly	Full backup of database should be done at least weekly, if not nightly.
Backup Transaction Log	Daily	Backing up the transaction log daily saves all committed transactions and clears the log.

Nightly Maintenance Script

The following script is the recommended nightly maintenance script:

```
UPDATE STATISTICS m7Object
go
UPDATE STATISTICS m7ObjectProperty
go
```


DBCC Commands

Use the following DBCC commands to perform maintenance tasks on the database:

```
DBCC CheckDB(mart)
go
DBCC CheckAlloc(mart)
go
DBCC CheckCatalog(mart)
go
```


Appendix B: Oracle Tuning Recommendations

This section contains the following topics:

[Recommended Disk Configurations](#) (see page 59)

[Database Configuration](#) (see page 60)

[Storage Parameters](#) (see page 66)

Recommended Disk Configurations

Oracle servers are available on many different platforms ranging from PCs to mainframes. To support the wide range of hardware platforms and application requirements, Oracle servers have a number of configuration options that you can use to tailor the behavior of an Oracle database. You can also use these configuration options to tailor your database. Some of the options documented in this guide only apply to specific Oracle versions. For further information, see the appropriate Oracle documentation.

If the Oracle server is not dedicated to CA ERwin MM, as database administrator you must be careful to balance the requirements of all users on the server. This is particularly true for database servers that support OLTP (On Line Transaction Processing) applications that require tight response characteristics.

The layout of Oracle database files on physical disk drives plays a major role in the performance of the database. When used with CA ERwin MM, Oracle performs large bursts of database operations (fetches, inserts, updates, and deletes) in a short period of time. This usage pattern often requires Oracle to perform a significant amount of disk I/O operations. For good performance, it is important that as many I/O operations as possible be performed in parallel. This enables Oracle to retrieve data faster during fetches and to write data faster during inserts, updates, and deletes.

To maximize parallel I/O access, split the Oracle database files across many physical disks. Ideally, you should place the database on four physical disks. If this is not possible, you can use three, two, or one disk systems.

Note: Installing Oracle on a single disk system causes significant performance degradation.

Database Configuration

Proper database configuration is essential to the smooth operation and good performance of an Oracle database. You can use the configuration options described to optimize the processing of your Oracle server.

Maximum Number of Processes

Oracle must be preconfigured at startup with a maximum number of processes that can access the database. The parameter that controls the maximum number of processes, called PROCESSES, is defined in the INIT.ora or INIT<SID>.ora file. It should be set to the maximum potential number of concurrent users plus seven. The additional seven are for Oracle background processes that must also access the database. Setting the number of processes at a lower value can prevent some users from accessing the server during peak usage times. Setting the number of processes at a higher value can waste a small amount of system memory resources.

Shared Pool

The shared pool is an area of Oracle memory that includes two main structures:

- The library cache, which stores parsed SQL and PL/SQL statements
- The dictionary cache, which stores the Oracle data dictionary (or Oracle metadata)

The SHARED_POOL_SIZE parameter is defined in the INIT.ora or INIT<SID>.ora file and is used to regulate the size of the shared pool. Set the SHARED_POOL_SIZE parameter to a minimum of 3500000 (3.5 million). Setting the shared pool to a smaller value can degrade performance, forcing Oracle to do disk I/O to retrieve objects that cannot fit in the shared pool. A larger shared pool may be required, depending on the size and number of models and the number of users.

Buffer Cache

The buffer cache serves as a memory cache for all data going to and from Oracle data files. When Oracle needs a block of data it first checks whether that block exists in the buffer cache. If it does, Oracle gets the data from the buffer cache—avoiding disk access. Having a large enough buffer cache lets the Oracle server bypass most I/O requests.

Buffer cache size is controlled by the `DB_BLOCK_BUFFERS` parameter defined in the `INIT.ora` or `INIT<SID>.ora` file. Set the buffer cache to a minimum of 4 MB. Values smaller than 4 MB force Oracle to do many more I/O requests and significantly degrades the performance. Values larger than 4 MB improve performance and should be used if memory is available.

Note: In Oracle, the `DB_BLOCK_BUFFERS` parameter is specified as a number of database blocks instead of actual size in bytes. To compute the value of `DB_BLOCK_BUFFERS`, divide the desired buffer cache size in bytes by the database block size defined by the `DB_BLOCK_SIZE` parameter in the `INIT.ora` or `INIT<SID>.ora` file.

Redo Log Files

Redo log files contain a record of all Data Manipulation Language (DML) commands (such as `INSERT`, `UPDATE`, and `DELETE` commands) performed on the database. As DMLs are performed, the Oracle engine writes them to sequential redo log files. Periodically, an Oracle background process retrieves the DMLs from the Redo log files and writes the actual changes to the Oracle tablespace files. This mechanism lets Oracle defer most of the I/O burden associated with DMLs to a background process that does not slow down the client processes.

Note: Oracle Redo log files are treated by the database engine as a ring. When one file fills up, the engine performs a log switch and starts writing to the next log file in the ring. When that log file fills, the engine switches again.

The number and size of the Redo log files is an important performance consideration. If the log files are too small or if there are not enough of them in the ring, Oracle may have to stall on a log switch. If this happens, the DMLs in the next log in the ring may not have been written to the tablespace files, and therefore the next log file has not been archived by the background `ARCH` process. For more information about Redo log archiving, see the Oracle documentation.

Redo log files are created when the database is created. However, Redo log files can be added or deleted at any time using Data Definition Language (DDL) statements.

You should have at least four Redo log files, each 2 MB in size. A smaller number of Redo log files or a smaller Redo log file size can cause I/O bottlenecks. If the disk space is available, using more than four Redo log files further reduces the chances of delayed log switches. Larger Redo log files improve performance, but care must be taken to adequately schedule checkpoints that write DMLs stored in the Redo logs to the tablespace files.

If the number of disks permits, you should mirror Redo log files by creating Redo log groups with two mirrored members per group. This offers the database some protection against single disk errors.

Increase Space in Redo Logs

The number and size of the Redo log files is an important performance consideration. Redo log files are created when the database is created.

To increase space in the Redo logs

1. Log in to Oracle using SQL*Plus as SYSDBA or SYSOPER and run the following script to create eight Redo logs that are 2 MB each:

```
rem -- pam1 -- temp dir

rem Generate creation script
SELECT 'ALTER DATABASE ADD LOGFILE "'
      || SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\', -1 , 1 ) )
      || 'REDO_11.LOG" SIZE 2M;'
from V$LOGFILE
where ROWNUM = 1
and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_11.LOG' )
UNION

SELECT 'ALTER DATABASE ADD LOGFILE "'
      || SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\', -1 , 1 ) )
      || 'REDO_12.LOG" SIZE 2M;'
from V$LOGFILE
where ROWNUM = 1
and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_12.LOG' )
UNION

SELECT 'ALTER DATABASE ADD LOGFILE "'
      || SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\', -1 , 1 ) )
      || 'REDO_13.LOG" SIZE 2M;'
from V$LOGFILE
where ROWNUM = 1
and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_13.LOG' )
UNION
```

```

SELECT 'ALTER DATABASE ADD LOGFILE "'
      || SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\', -1 , 1 ) )
      || 'REDO_14.LOG" SIZE 2M;'
from V$LOGFILE
where ROWNUM = 1
and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_14.LOG' )
UNION

SELECT 'ALTER DATABASE ADD LOGFILE "'
      || SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\', -1 , 1 ) )
      || 'REDO_15.LOG" SIZE 2M;'
from V$LOGFILE
where ROWNUM = 1
and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_15.LOG' )
UNION

SELECT 'ALTER DATABASE ADD LOGFILE "'
      || SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\', -1 , 1 ) )
      || 'REDO_16.LOG" SIZE 2M;'
from V$LOGFILE
where ROWNUM = 1
and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_16.LOG' )
UNION

SELECT 'ALTER DATABASE ADD LOGFILE "'
      || SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\', -1 , 1 ) )
      || 'REDO_17.LOG" SIZE 2M;'
from V$LOGFILE
where ROWNUM = 1
and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_17.LOG' )
UNION

SELECT 'ALTER DATABASE ADD LOGFILE "'
      || SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\', -1 , 1 ) )
      || 'REDO_18.LOG" SIZE 2M;'
from V$LOGFILE
where ROWNUM = 1
and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_18.LOG' )

.
Spool &1.AddLog.ORA
/
Spool Off
COMMIT ;

SELECT 'ALTER SYSTEM SWITCH LOGFILE ;' from V$LOG where ROWNUM < 5
.
Spool &1.SwtchLog.ORA
/
Spool Off
COMMIT ;

```

```
rem Generate Deletion script
SELECT 'ALTER DATABASE DROP LOGFILE GROUP ' || TO_CHAR ( GROUP# ) || ',' from V$LOG
where BYTES < 2097152

.
Spool &1.DropLog.ORA
/
Spool Off
COMMIT ;

@&1.AddLog.ORA
$Del &1.AddLog.ORA
COMMIT ;

@&1.SwtchLog.ORA
$Del &1.SwtchLog.ORA
COMMIT ;

@&1.DropLog.ORA
$Del &1.DropLog.ORA
COMMIT ;
```

2. Run the following query to verify the new Redo log configuration:

```
SQLWKS> select group#, status, bytes from v$log;
```

Note: Oracle does not let you drop an older Redo log file that is still ACTIVE. If you must drop an older Redo log file that is still ACTIVE, drop that log file manually.

Redo Log Buffer

The Redo log buffer is an area in memory that Oracle uses to collect DMLs before they are written to the Redo log files. The log file write occurs when either a transaction commits or a Redo log buffer is full. Because CA ERwin MM tends to generate fairly large transactions, use a log buffer size of 163840 bytes. (For single disk Oracle installations, use a log buffer size of 655360 bytes.) Setting the Redo log buffer size to a value smaller than the recommended value can degrade I/O performance. Using a larger value requires more memory.

The Redo log buffer size is defined using the LOG_BUFFER parameter in the INIT.ora or INIT<SID>.ora file.

Checkpoints

An Oracle checkpoint is an event that posts DMLs from the Redo log files to the tablespace files. Checkpoints always occur after a Redo log file switch and can also be configured to occur at predefined time intervals. For CA ERwin MM, which generates large transactions, checkpoints should occur only after log switches to minimize I/O.

To ensure a checkpoint only after a log switch, *do not* set the LOG_CHECKPOINT_INTERVAL and LOG_CHECKPOINT_TIMEOUT parameters in the INIT.ora or INIT<SID>.ora file.

Turn on the Optimizer Mode

You can increase the response time of a CA ERwin MM mart on an Oracle 9.x system, when you add the optimizer mode in the initialization file.

Note: This is not needed on an Oracle 10g system.

To add the optimizer mode in the initialization file:

1. Open the init.ora file located in \oracle\admin\dbname\pfile\ where dbname is the name of the database in which mart is created.

The init.ora file is open.

2. Add *optimizer_mode=ALL_ROWS* in the optimizer section.

The optimizer mode is added.

3. Shutdown and restart the database.

Then log into the CA ERwin MM database as sysdba

4. Using SQL*Plus, execute the following commands:

```
execute dbms_stats.gather_system_stats('Start');
execute dbms_stats.gather_system_stats('Stop');
EXEC DBMS_STATS.gather_database_stats;
```

The commands are executed and the optimizer mode is enabled.

Storage Parameters

Object storage parameters in Oracle determine the amount of space allocated for each object in the database. Setting these parameters correctly is critical to both operation and performance of the database. Incorrect storage allocations can cause a database object to run out of space, which prevents you from saving models to the repository. Inefficient selection of storage parameters can lead to performance problems by forcing Oracle to do time-consuming space management operations during DML statement execution.

Recommendations for storage parameter values differ based on the size of your model:

- Small data models: 1 to 50 Entities
- Medium data models: 51 to 100 Entities
- Large data models: over 100 Entities

Rollback Segments

Rollback segments contain undo information for all changes performed by noncommitted transactions. Rollback segments are a shared resource used by all active transactions in the database. When a transaction starts, Oracle binds that transaction to a particular rollback segment. As DMLs in the transaction execute, rollback segment space is used. For large transactions, rollback segments may need to allocate new extents as the transaction continues. When the transaction ends, a properly configured Oracle database releases the additional rollback extents so that they can be used for other rollback segments. To ensure that the additional rollback extents are released, configure the OPTIMAL parameter for each rollback segment.

Use a maximum of five transactions per rollback segment by setting the TRANSACTIONS_PER_ROLLBACK_SEGMENT parameter in the INIT.ora or INIT<SID>.ora file. Use the following rollback segment storage parameters for small, medium, and large models:

Model Type	Initial Extent Size	Next Extent Size	Optimal Size
Small	1 MB	1 MB	~ 2 MB x (# of transactions)
Medium	6 MB	6 MB	~ 12 MB x (# of transactions)
Large	6 MB	6 MB	~ 12 MB x (# of transactions)

Note: The “# of transactions” is the maximum number of simultaneous CA ERwin MM connections to Oracle.

In Oracle it is common to see the following error statement when the Rollback logs are not set up optimally:

ORA-01562 failed to extend rollback segment number string

Set Optimal Configuration for the Rollback Logs

For large transactions, rollback segments may need to allocate new extents as the transaction continues. When the transaction ends, a properly configured Oracle database releases the additional rollback extents so that they can be used for other rollback segments. To ensure that the additional rollback extents are released, configure the OPTIMAL parameter for each rollback segment.

To set the optimal configuration for the rollback logs of large databases

1. Log in to Oracle as SYS or SYSTEM and run the following query and check the result to view the current configuration of the Rollback Segments:

```
SQLWKS> select SEGMENT_NAME, INITIAL_EXTENT, NEXT_EXTENT, MIN_EXTENTS,
MAX_EXTENTS, STATUS from dba_rollback_segs;
```

The result is:

SEGMENT_NAME	INITIAL_EX	NEXT_EXTEN	MIN_EXTENT	MAX_EXTENT	STATUS
SYSTEM	51200	51200	2	121	ONLINE
RB_TEMP	102400	102400	2	121	OFFLINE
RB1	2097152	2097152	2	121	ONLINE
RB2	2097152	2097152	2	121	ONLINE
RB3	2097152	2097152	2	121	ONLINE
RB4	2097152	2097152	2	121	ONLINE
RB5	2097152	2097152	2	121	ONLINE
RB6	2097152	2097152	2	121	ONLINE
RB7	2097152	2097152	2	121	ONLINE
RB8	2097152	2097152	2	121	OFFLINE
RB9	2097152	2097152	2	121	OFFLINE
RB10	2097152	2097152	2	121	OFFLINE
RB11	2097152	2097152	2	121	OFFLINE
RB12	2097152	2097152	2	121	OFFLINE
RB13	2097152	2097152	2	121	OFFLINE
RB14	2097152	2097152	2	121	OFFLINE
RB15	2097152	2097152	2	121	OFFLINE
RB16	2097152	2097152	2	121	OFFLINE

18 rows selected.

Note: In this example, there are sixteen rollback segments with 2 MB INITIAL EXTENT, 2 MB NEXT EXTENT, and MAX EXTENTS of 121 MB.

2. Run the following query for each Rollback Segment:

```
alter rollback segment rb1 offline;  
drop rollback segment rb1;  
create public rollback segment rb1  
tablespace rollback_data  
storage (initial 6M  
next 6M  
minextents 2  
maxextents 121  
optimal 12M);  
alter rollback segment rb1 online;  
alter rollback segment rb2 offline;  
drop rollback segment rb2;  
create public rollback segment rb2  
tablespace rollback_data  
storage (initial 6M  
next 6M  
minextents 2  
maxextents 121  
optimal 12M);  
alter rollback segment rb2 online;
```

<Repeat for each Rollback Segment>

This query sets the optimal Rollback Segment configuration to 6 MB INITIAL EXTENT, 6 MB NEXT EXTENT, 2 MB MIN EXTENT, 121 MB MAX EXTENT, and 12 MB OPTIMAL.

3. Run the following query to verify the Rollback segment configuration changes:

```
SQLWKS> select SEGMENT_NAME, INITIAL_EXTENT, NEXT_EXTENT,
MIN_EXTENTS, MAX_EXTENTS, STATUS from dba_rollback_segs;
```

The result is:

SEGMENT_NAME	INITIAL_EXTENT	NEXT_EXTENT	MIN_EXTENTS	MAX_EXTENTS	STATUS
SYSTEM	51200	51200	2	121	ONLINE
RB_TEMP	102400	102400	2	121	OFFLINE
RB1	6291456	6291456	2	121	ONLINE
RB2	6291456	6291456	2	121	ONLINE
RB3	6291456	6291456	2	121	ONLINE
RB4	6291456	6291456	2	121	ONLINE
RB5	6291456	6291456	2	121	ONLINE
RB6	6291456	6291456	2	121	ONLINE
RB7	6291456	6291456	2	121	ONLINE
RB8	6291456	6291456	2	121	ONLINE
RB9	6291456	6291456	2	121	ONLINE
RB10	6291456	6291456	2	121	ONLINE
RB11	6291456	6291456	2	121	ONLINE
RB12	6291456	6291456	2	121	ONLINE
RB13	6291456	6291456	2	121	ONLINE
RB14	6291456	6291456	2	121	ONLINE
RB15	6291456	6291456	2	121	ONLINE
RB16	6291456	6291456	2	121	ONLINE

18 rows selected.

Tablespaces

To improve performance you can create more than one tablespace for indexes. Then, after you install the software, you can move some indexes to the other tablespaces.

The following table lists the suggested initial sizes for DATA and INDEX tablespaces:

Tablespace	Small	Medium	Large
DATA	200 MB	500 MB	1 GB
INDEX	200 MB	250 MB	500 MB

DATA and INDEX tablespaces with these initial sizes can accommodate at least ten models or versions.

Recommended Initial Table Sizes

The following table lists the recommended initial extent sizes in kilobytes for the most important tables for small, medium, and large models:

Recommended Initial Extent Size (KB)			
Table Name	Small	Medium	Large
m7Object	1096	2193	10963
m7ObjectProperty	3288	6579	32889

These initial extent sizes are chosen so they can accommodate two versions of a model. These values are the recommended values of the INITIAL parameter in the STORAGE clause of the CREATE TABLE statement. The size of the next extent (NEXT parameter in the STORAGE clause) should be half of the corresponding initial size.

Recommended Initial Indexes

The following lists the recommended initial extent sizes in kilobytes for the most important indexes for small, medium, and large models:

Recommended Initial Extent Size (KB)			
Index Name	Small	Medium	Large
XPkm7Object	1415	2829	14146
XPkm7ObjectProperty	2098	4195	20975

Appendix C: CA ERwin MM Performance

This section contains the following topics:

[Optimize Your CA ERwin MM Performance](#) (see page 71)

[Reindex the Database](#) (see page 72)

[Run the Database Statistics](#) (see page 77)

Optimize Your CA ERwin MM Performance

The performance of CA ERwin MM depends on many factors. This appendix outlines some steps you can take to improve your performance. The five main components where a slow down can occur are the database, the server, the network, the client PC, and the CA ERwin MM software.



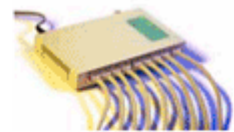
The Database.

- Reindexing.
- Statistics.
- Tuning.



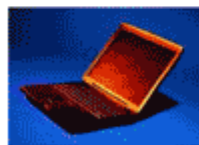
The Server.

- RAM.
- Speed (GHz).
- CPU.
- Disk space.
- Dedicated to AF Model Manager or shared by multiple applications?



The Network.

- LAN or WAN?
- Speed.
- Throughput.
- Traffic.



The Client Machine.

- RAM.
- Speed (GHz).
- CPU.
- Disk space & Virtual Memory.



The Software.

You should regularly reindex the database and run the database statistics. Sometimes running the database statistics alone does not affect the performance, however it is good practice to run the database statistics on a regular basis to ensure the maximum performance of your database.

In past releases, multiple versions of the same model in the mart can slow down overall performance. You can remove versions by selectively purging them using the MM Version Manager. CA ERwin MM r7.2 or later lets you to disable the versioning so it does not create a new version with every save of a model to the mart.

If you have many domains in a model, they consume a lot of memory/file space. Having too many domains can slow down both CA ERwin Data Modeler and CA ERwin MM. If you need to use many domains, we suggest you create a template model to house them, and then import the specific domains that you need, to each of your individual models. If you are using the domains to enforce unique attribute or column naming standards, you can alternatively use the ERwin Glossary (*.nsm) for that purpose.

Check the CA ERwin MM Release Notes for the latest system requirements. Greater CPU speed, and more RAM and disk space result in better performance.

Reindex the Database

Over a period of time, database indexes become fragmented. A large number of inserts and deletes can lead to significant performance degradation. The scripts provided in this appendix rebuild the CA ERwin MM indexes and fix them. In some cases, it can make a significant difference in the CA ERwin MM performance.

We recommend that you reindex your CA ERwin MM database nightly or during off-peak hours. As the index is recreated, the process can temporarily degrade the performance of your database. You can create an automated batch job to do the reindex. You should also reindex the database after a large model merge, a save to CA ERwin MM, and right after a conversion of the CA ERwin MM mart when upgrading to a later version of the software.

Run the Oracle DBMS Reindex Script

As the new index is being built, it coexists with the old index in the database. For this reason, you should plan for enough space to store both the old index and the new index. When the index is rebuilt, the new index becomes available, the old index is dropped, and the space is reclaimed by the database. If you encounter any errors while rebuilding the indexes, re-run the statements. If you require more space to rebuild those specific indexes, add more storage to your index tablespace, and then try rebuilding those specific indexes again.

To run the Oracle DBMS reindex script:

1. Create the mmreindex.ora script and copy it locally.
2. Edit the script and replace 'MODELMART' with the name of the CA ERwin MM schema-owner and 'MMINDEX' with the name of the CA ERwin MM index tablespace. Save your changes.
3. Connect to SQL*PLUS as the user SYS.
4. Grant the 'ALTER ANY INDEX' privilege to the CA ERwin MM Schema Owner.
5. Run your Oracle query tool and execute the following at the SQL prompt:
`GRANT ALTER ANY INDEX TO <MART SCHEMA OWNER>;`
6. Disconnect user SYS and Connect to your Oracle query tool as the Schema Owner.
7. Execute the following Script at the SQL Prompt,
`@c:\mmreindex.ora`

Example: Oracle DBMS Reindex Script

An example of the Oracle dbms reindex script (MMReIndex.ora):

-
- Object: MMReIndex.ora
 - Desc: Use this Procedure to ReIndex the MM ORACLE Repository whenever a Merge/Save of big model is done to MM
 - Limitation(s) is specific to Oracle Releases >= 817
 - For ORACLE DBMS < 8i Modify the script to Use NOPARALLEL
 - NOTE: You will need to change MODEL MART to the Model Manager schema owner name.
 - You will need to change MMINDEX to the Model Manager index tablespace.
 - Oracle indexes are not self-balancing. They become fragmented after a large number of INSERTs and DELETE which may lead to significant performance degradation.
 - This script rebuilds the Model Manager indexes and cures them.
-

```
set pagesize 1000
set linesize 2000
set verify off
set feedback off
set heading off
spool c:\mmreindex.ora
SELECT      'ALTER INDEX ' || USER || '.' || INDEX_NAME ||
'REBUILD PARALLEL NOLOGGING COMPUTE STATISTICS TABLESPACE MMINDEX;'
FROM    DBA_INDEXES
WHERE OWNER = UPPER('MODEL MART')
AND    (INDEX_NAME like 'XPK%' or INDEX_NAME like 'XAK%'
        or INDEX_NAME like 'XIE%')
order by index_name;
spool off
set heading on
set pagesize 24
set verify on
set feedback on
@c:\mmreindex.ora
/
```

Example: Microsoft SQL Server DBMS Reindex Script

An example of the MS SQL Server dbms reindex script:

```
-- Drop the Procedure appropriately
IF EXISTS (SELECT name FROM sysobjects WHERE name = N'usp_ReIndex' AND type = N'P')
Begin
    DROP PROCEDURE usp_ReIndex
    Print 'Procedure Dropped'
End
GO
```

```
-- Object:  usp_ReIndex
-- Desc:    Use this Procedure to ReIndex the MM SQL REpository whenever a
--          Merge/Save of big model is done to MM
-- Change History:
-- Name      Date      Reason
```

```
CREATE PROCEDURE usp_ReIndex AS
Declare
    @Cmd  varchar(2000),
    @Name Sysname
DECLARE tmp_Reindex CURSOR LOCAL FOR
    SELECT Name
    FROM  SysObjects
    WHERE Type = 'U'
OPEN tmp_Reindex
FETCH NEXT FROM tmp_Reindex INTO @Name
WHILE @@FETCH_STATUS = 0
BEGIN
    Print 'Processing Index for Table ' + @Name
    Set @cmd = 'DBCC DBREINDEX (' + @Name + ', "", 0)'
    Exec (@Cmd)
    If @@Error <> 0
        Print 'Error Reindexing Table ' + @Name
    FETCH NEXT FROM tmp_Reindex INTO @Name
END
Close tmp_Reindex
go
-- ReIndex the DB
Exec usp_ReIndex
```

Example: Sybase DBMS Reindex Script

An example of the Sybase dbms reindex script (MMReIndex.ora):

```
-- Drop the Procedure appropriately
IF EXISTS (SELECT name FROM sysobjects WHERE name = N'usp_ReIndex' AND type = N'P')
Begin
    DROP PROCEDURE usp_ReIndex
    Print 'Procedure Dropped'
End
GO
```

```
-- Object: usp_ReIndex
-- Desc: Use this Procedure to ReIndex the MM SQL REpository whenever a
-- Merge/Save of big model is done to MM
-- Change History:
-- Name Date Reason
```

```
CREATE PROCEDURE usp_ReIndex AS
Declare
    @Cmd varchar(2000),
    @Name Sysname(100) ,
    @output_str varchar( 255 )
DECLARE tmp_Reindex CURSOR FOR
    SELECT name
    FROM sysobjects
    WHERE type = 'U'
OPEN tmp_Reindex
FETCH tmp_Reindex INTO @Name
WHILE ( @@sqlstatus = 0 )
BEGIN
    SELECT @output_str = 'Processing Index for Table ' + @Name
    Print @output_str
    --Set @Cmd = 'DBCC REINDEX (' + @Name + ')'
    --Exec (@Cmd)
    DBCC REINDEX (@Name )
    If @@Error <> 0
    Begin
        SELECT @output_str = 'Processing Index for Table1 ' + @Name
        Print @output_str
    end
    FETCH tmp_Reindex INTO @Name
END
Close tmp_Reindex
go
EXEC usp_ReIndex
```

Troubleshooting the Reindexing Script

If your query tool responds:

no rows selected
not spooling currently

it means you do **not** have the correct name for the CA ERwin MM schema-owner. You must get the correct user name, replace 'MODELMART' with the user name, and re-run the script.

Run the Database Statistics

Database statistics show the distribution of the data in the database, and how the data is stored. When a database executes a query, it uses an optimizer to determine the best path to access the data. The optimizer relies on execution plans that specify the order in which the database accesses the tables and the exact steps used to pull the data. The database bases the creation of the execution plans on the database statistics.

As the data grows and changes, the statistics quickly become outdated and no longer reflect the true condition of the database. As a result, the execution plans no longer apply and the optimizer makes poor decisions when processing queries. You should run statistics regularly to ensure the maximum performance of your database and, therefore, your CA ERwin MM mart. We recommend that you run the database statistics nightly, and create an automated batch job to simplify the process.

Example: Statistics Commands (Oracle)

NOTE: In the examples below, the schema owner name is 'MODELMART'

To collect statistics for the CA ERwin MM (does not include the index statistics):

```
EXEC DBMS_STATS.gather_schema_stats ('MODELMART',DBMS_STATS.AUTO_SAMPLE_SIZE);
```

To collect statistics for the CA ERwin MM (includes the index statistics):

```
EXEC DBMS_STATS.gather_schema_stats (ownname => 'MODELMART',  
cascade =>true, estimate_percent => dbms_stats.auto_sample_size);
```

To delete the statistics:

```
EXEC DBMS_STATS.delete_schema_stats ('MODELMART');
```

Example: Statistics Script (MS SQL Server)

```
update statistics m7Access with sample 50 percent
update statistics m7Action with sample 50 percent
update statistics m7ActionGrouping with sample 50 percent
update statistics m7Application with sample 50 percent
update statistics m7BranchLog with sample 50 percent
update statistics m7Class with sample 50 percent
update statistics m7Control with sample 50 percent
update statistics m7ControlCache with sample 50 percent
update statistics m7Deleted with sample 50 percent
update statistics m7Library with sample 50 percent
update statistics m7License with sample 50 percent
update statistics m7Lock with sample 50 percent
update statistics m7LockType with sample 50 percent
update statistics m7Master with sample 50 percent
update statistics m7Object with sample 50 percent
update statistics m7ObjectProperty with sample 50 percent
update statistics m7OldVersions with sample 50 percent
update statistics m7Permission with sample 50 percent
update statistics m7PermissionProfile with sample 50 percent
update statistics m7PermissionType with sample 50 percent
update statistics m7ProfileAssignment with sample 50 percent
update statistics m7Property with sample 50 percent
update statistics m7Query with sample 50 percent
update statistics m7Session with sample 50 percent
update statistics m7User with sample 50 percent
```

Index

A

administrator
tasks • 33, 46

B

buffer cache • 61

C

checkpoints • 65
create
 an Oracle user • 27
 database • 21
 tablespace • 27

D

database
 configuration • 60
 convert • 39, 40, 41, 46
 create • 15, 28
 database and log sizing • 55
 initialize • 15, 33, 46
 maintenance • 55
 manage • 33, 46
 options • 54
 placement • 52
 post creation tasks • 20
 tuning parameters • 53
 update • 46
 upgrade recommendations • 42

I

install
 administrative files • 30
 and initialize CA ERwin MM • 15
 CA ERwin MM Objects • 25
 software upgrade • 44

M

Microsoft SQL Server
 connections • 18
 conversion • 46
 create • 18
 database creation requirements • 28
 database placement • 52

 initialize • 33, 43, 46
 maintenance • 55
 server tuning recommendations • 51
 settings • 16
 upgrade recommendations • 43
modeling, workgroup • 9, 11, 12, 13

O

Oracle server
 connections • 26
 conversion • 43, 46
 create a user • 27
 create the database • 22
 initialize • 33
 Oracle SQL commands • 27
 Transact SQL commands • 21
 tuning recommendations • 59
 upgrade recommendations • 43

R

recommendations
 disk • 59
 index • 70
 table size • 70
requirements
 system • 28

S

Sybase server
 connections • 20
 conversion • 46
 create database • 19
 initialize • 33
 server tuning recommendations • 51
 upgrade recommendations • 43

T

tablespaces • 69
transaction log • 53

U

upgrade
 CA ERwin MM • 39
 recommendations • 42
 software • 39
